**Exercise 3**

Entity: BOOK

B\_ID (PK)

Author (FK; inked to PERSON)

Translator (FK; linked to PERSON)

Note

Publisher (FK; linked to PUBLISHER)

Year

Dimension

Language (FK; linked to LANGUAGE)

Entity: PERSON

P\_ID (PK)

Name\_first

Name\_last

Year\_of\_birth

Year\_of\_death

Entity: PUBLISHER

PUBL\_ID (PK)

Name

Entity: LANGUAGE

Lang\_CODE (PK)

Language

Entity: LETTER

Letter\_ID (PK)

Sender (FK; linked to PERSON)

Recipient (FK; linked to PERSON)

Transcription

Entity: REFERENCE

REF\_ID (PK)

Letter (FK; linked to LETTER)

Book (FK; linked to BOOK)

**Exercise 6**

1. The last name, the first name and the year of birth of all the persons in the table CREATOR. Sort the rows by year of birth. The most recent year should appear first.

SELECT NAME\_LAST, NAME\_FIRST, YEAR\_OF\_BIRTH  
FROM CREATOR  
ORDER BY YEAR\_OF BIRTH DESC

1. All the columns in the table CITY, and sort the rows alphabetically, by name of the city.

SELECT \*  
FROM CITY  
ORDER BY NAME

1. The treasures that were produced after 1800. Display only the title and the year

SELECT TITLE, YEAR  
FROM TREASURE  
WHERE YEAR > 1800

1. The creators whose first name starts with the letter "W"

SELECT NAME\_LAST, NAME\_FIRST   
FROM CREATOR  
WHERE NAME\_FIRST LIKE ‘%W’

1. The distinct subject codes in the table TREASURE, and sort the result alphabetically.

SELECT DISTINCT SUBJECT  
FROM TREASURE  
ORDER BY SUBJECT

1. The year of the oldest treasure in the database.

SELECT MIN(YEAR)  
FROM TREASURE

1. The number of countries in the table CREATOR.

SELECT COUNT( DISTINCT COUNTRY\_BORN )  
FROM CREATOR

1. The most recent year of death in the table CREATOR.

SELECT MAX(YEAR\_OF\_DEATH)  
FROM CREATOR

1. The oldest treasure from each library. For this query, you only need to work with the identifiers of the libraries and not with their full names.

SELECT LIBRARY, MIN(YEAR)  
FROM TREASURE  
GROUP BY LIBRARY

1. The number of persons for each country in the table CREATOR.

SELECT COUNTRY\_BORN , COUNT(\*)  
FROM CREATOR  
GROUP BY COUNRY\_BORN

1. Those libraries that are represented by two or more treasures.

SELECT LIBRARY , COUNT(\*)  
FROM TREASURE  
GROUP BY LIBRARY  
HAVONG COUNT(\*) > 2

1. Those subjects that have been assigned to two or more treasures in the database.

SELECT SUBJECT , COUNT(\*)  
FROM TREASURE  
GROUP BY SUBJECT  
HAVING COUNT(\*) >= 2

1. The title, year, and provide a full description of the subject code of all the treasures in the database.

SELECT TITLE, YEAR, SUBJECT.SUBJECT  
FROM TREASURE T, SUBJECT S  
WHERE T.SUBJECT = S.CODE

1. The names of all the libraries and the names of the cities in which they are located.

SELECT L.NAME , C.NAME  
FROM LIBRARY L, CITY C  
WHERE L.CITY = C.CITY\_CODE

1. A list with all names of all cities and the full names of the countries in which they are situated.

SELECT CITY.NAME , COUNTRY.NAME  
FROM CITY, COUNTRY  
WHERE COUNTRY = COUNTRY\_CODE

1. The title, year, and a full description of the subject code of all treasures, and the full names of their creators. Sort the list alphabetically by the last name of the creator.

SELECT TITLE, YEAR S,NAME, NAME\_FIRST, NAME\_LAST  
FROM TREASURE T, SUBJECT S, CREATOR C

WHERE T.SUBJECT = S.CODE

AND T.CREATOR = CREATOR\_ID

**Exercise 7**

1. Which authors have written more than one beststeller?

SELECT FIRST\_NAME , LAST\_NAME , COUNT(\*)

FROM BOOK, AUTHOR

WHERE AUTHOR = AUTHOR\_ID

GROUP BY AUTHOR

HAVING COUNT(\*) > 1

1. Which publishers are responsible for more than three bestsellers?

SELECT PUBLISHER.NAME , LAST\_NAME , COUNT(\*)

FROM BOOK, PUBLISHER

WHERE PUBLISHER = PUBLISHER.NAME

GROUP BY PUBLISHER

HAVING COUNT(\*) > 2

1. What are the best-selling books in the genre “Popular Science”?

SELECT TITLE, SOLD

FROM BOOK, GENRE

WHERE BOOK.GENRE = GENRE.GENRE\_ID  
AND GENRE.GENRE = ‘Popular Science’

ORDER BY SOLD

1. What is the average number of sales of a book written by J.K. Rowling?

SELECT AVG( SOLD )  
FROM BOOK, AUTHOR

WHERE AUTHOR = AUTHOR\_ID

AND LAST\_NAME = ‘Rowling’

1. For each genre, show the average number of sales

SELECT GENRE.GENRE, AVG(\*)

FROM BOOK, GENRE

WHERE BOOK.GENRE = GENRE.GENRE\_ID

1. For each genre, show the maximum number of books that have been sold

SELECT GENRE.GENRE, MAX(\*)

FROM BOOK, GENRE

WHERE BOOK.GENRE = GENRE.GENRE\_ID

1. For each author, show the total number of sales

SELECT FIRST\_NAME + ‘’ + LAST\_NAME , SUM( SOLD )

FROM BOOK, AUTHOR

WHERE AUTHOR = AUTHOR\_ID

GROUP BY AUTHOR

1. For each year, show the total number of bestsellers

SELECT YEAR , COUNT( \* )

FROM BOOK  
GROUP BY YEAR